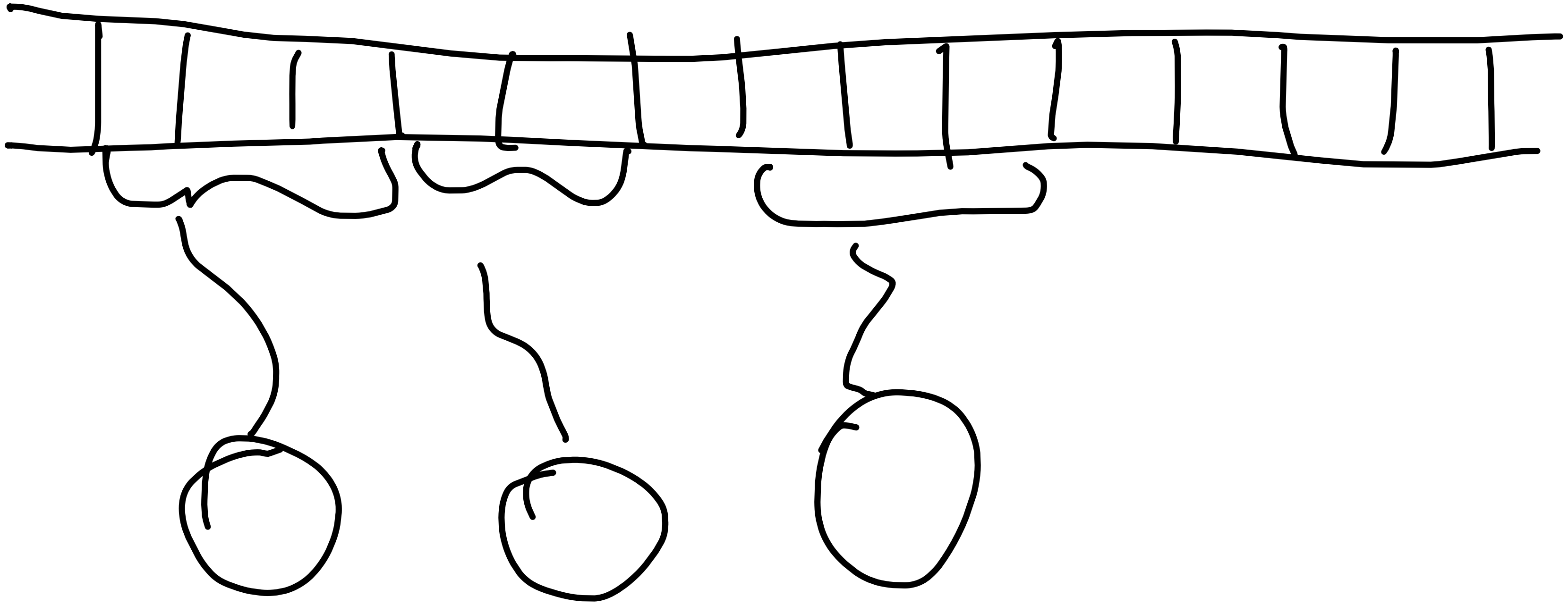
Wireless communication

algo21

Radio channel access

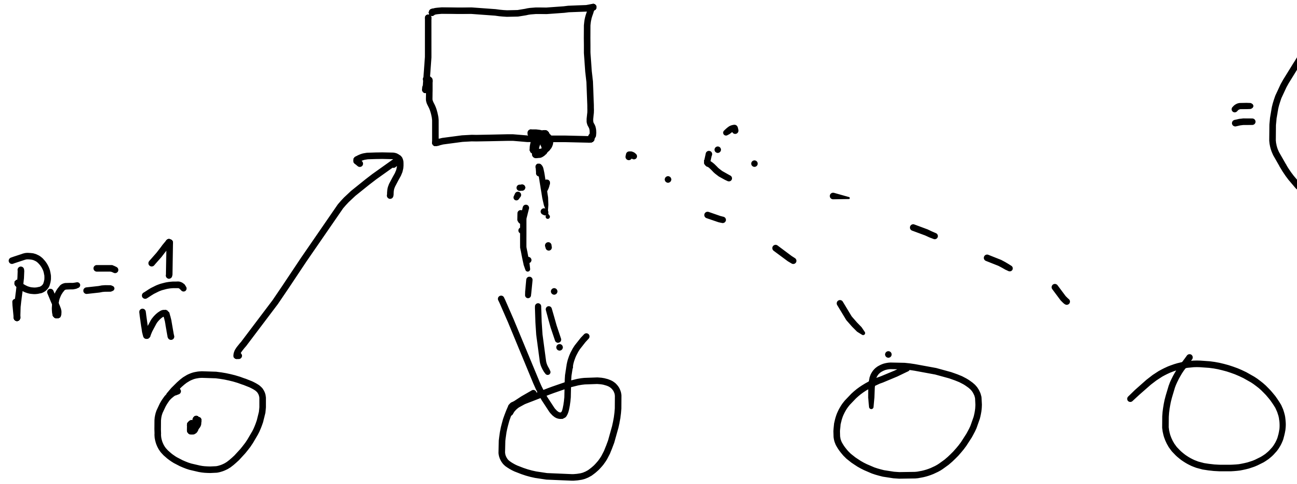
TDMA



Algorithm 12.1 Slotted Aloha

Every node v executes the following code:
repeat
    transmit with probability $1/n$
until one node has transmitted alone

$$\Pr(1 \text{ transmits}) = \binom{n}{1} \cdot \frac{1}{n} \cdot \left(1-\frac{1}{n}\right)^{n-1} = \approx n \cdot \cancel{\frac{1}{n}} \cdot \frac{1}{e}$$



$$Pr[X = 1] = n \cdot \frac{1}{n} \cdot \left(1 - \frac{1}{n}\right)^{n-1} \approx \frac{1}{e},$$

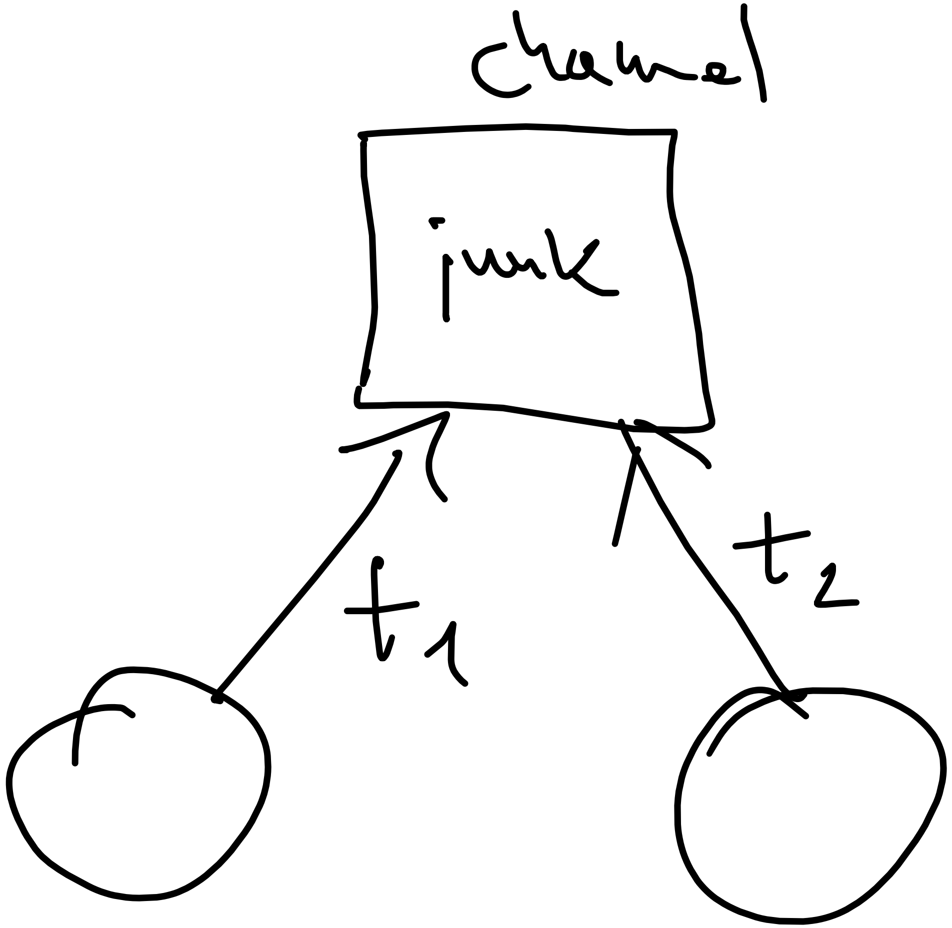
Initialization

devices, no knowledge on
who joined the network

⇒ assign them numbers $1, \dots, n$

LE - leader ~ 1
rest: LE leader - 2
rest: LE leader - 3
⋮

Collision detection

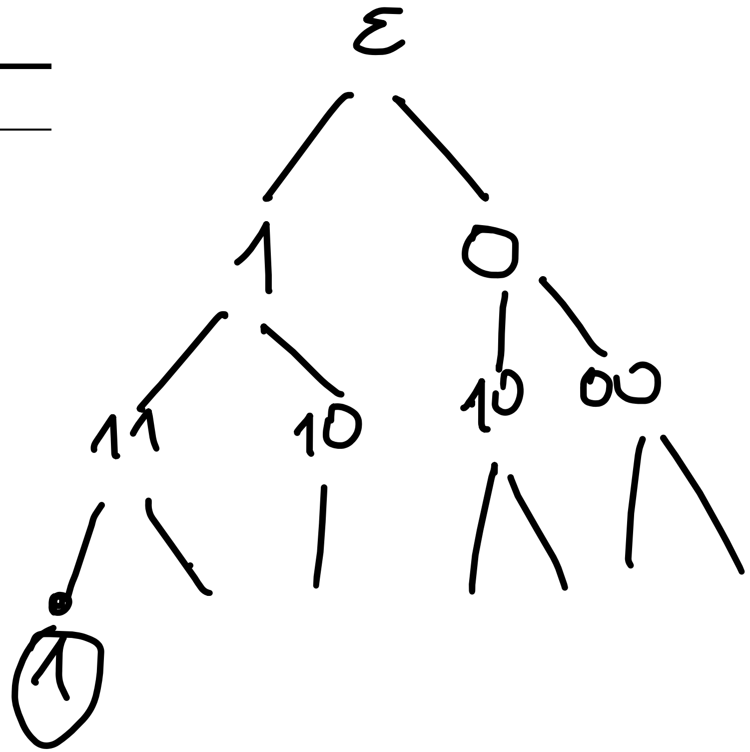


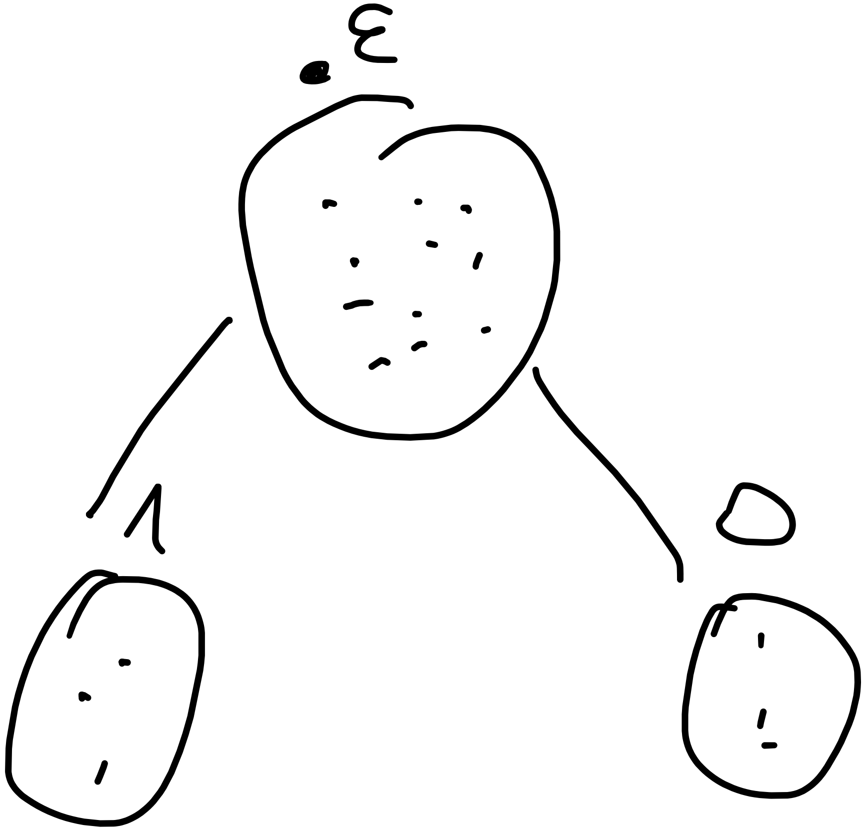
1) junk

2) no readable but ≥ 2 transmit

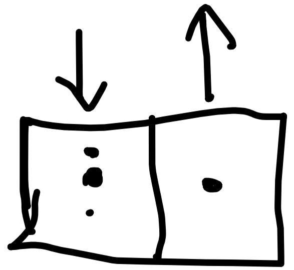
Algorithm 12.5 Initialization with Collision Detection

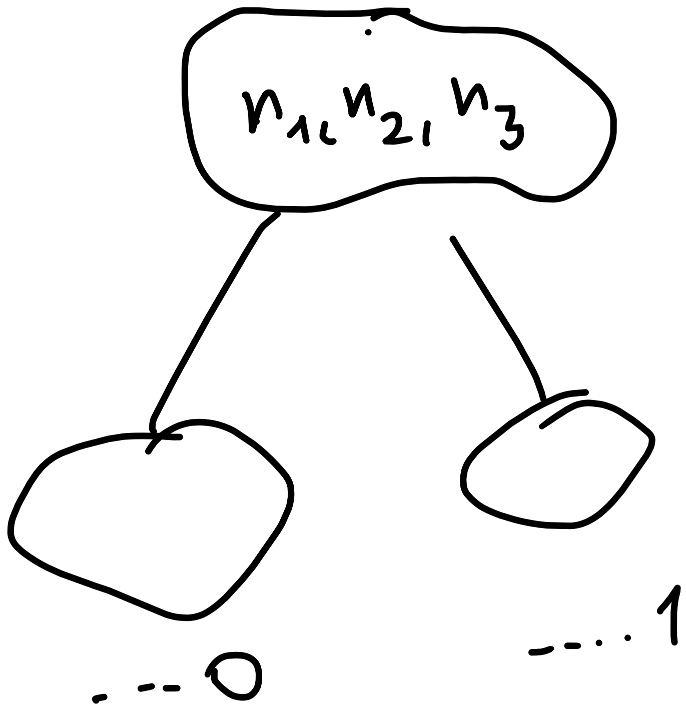
```
Every node v executes the following code:
nextId := 0
myBitstring := ‘’                      ◁ initialize to empty string
bitstringsToSplit := [‘’]              ◁ a queue with sets to split

while bitstringsToSplit is not empty do
    b := bitstringsToSplit.pop()

    repeat
```




ε
1
0

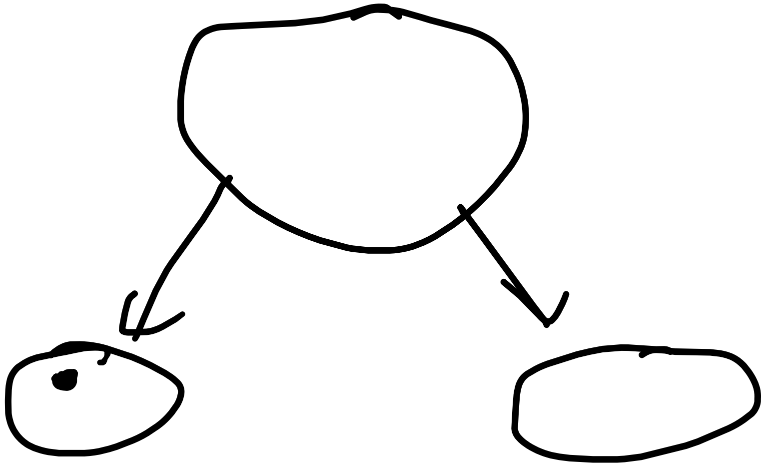
```
repeat
    if b = myBitstring then
        choose r uniformly at random from {0, 1}
        in the next two time slots:
        transmit in slot r, and listen in other slot
    else
        it is not my bitstring, just listen in both slots
    end if
until there was at least 1 transmission in both slots
if b = myBitstring then
    myBitstring := myBitstring ‖ r        ◁ append bit r
end if
```

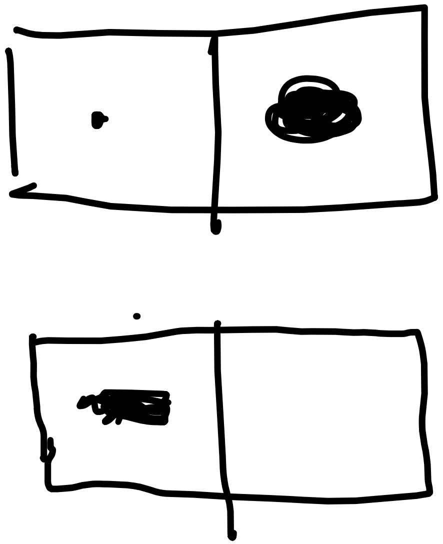




```
repeat
    if b = myBitstring then
        choose r uniformly at random from {0, 1}
        in the next two time slots:
        transmit in slot r, and listen in other slot
    else
        it is not my bitstring, just listen in both slots
    end if
until there was at least 1 transmission in both slots
if b = myBitstring then
    myBitstring := myBitstring + r          ◁ append bit r
end if
```

```
    for r ∈ {0, 1} do
        if some node u transmitted alone in slot r then
            node u becomes ID nextId and becomes passive
            nextId := nextId + 1
        else
            bitstringsToSplit.push(b + r)
        end if
    end for
end while
```





```
    for r ∈ {0, 1} do
        if some node u transmitted alone in slot r then
            node u becomes ID nextId and becomes passive
            nextId := nextId + 1
        else
            bitstringsToSplit.push(b + r)
        end if
    end for
end while
```

Theorem 12.6. *Algorithm 12.5 correctly initializes n nodes in expected time $\mathcal{O}(n)$.*

Initialization with no collision detection but with a leader

S = nodes that transmit

	nodes in S transmit	nodes in $S \cup \{\ell\}$ transmit
$\lvert S\rvert = 0$	✗	✔
$\lvert S\rvert = 1, S = \{\ell\}$	✔	✔
$\lvert S\rvert = 1, S \neq \{\ell\}$	✔	✗
$\lvert S\rvert \geq 2$	✗	✗

Table 12.7: Using a leader to distinguish between noise and silence: ✗ represents noise/silence, ✔ represents a successful transmission.

No CD:

transmission, noise

Uniform Leader election

- single
- aware

Algorithm 12.10 Uniform leader election

```
1: Every node v executes the following code:
2: for k = 1, 2, 3, ... do
3:     for i = 1 to ck do
4:         transmit with probability p := 1/2^k
5:         if node v was the only node which transmitted then
6:             v becomes the leader
7:             break
8:         end if
9:     end for
10: end for
```

$c \cdot k$

$n \approx 2^k$

$p = \frac{1}{n} \cdot 2$

$\binom{n}{1} \cdot \frac{2}{n} \cdot \left(1 - \frac{2}{n}\right)^{n-1}$

$\approx \frac{1}{e^2} \approx$

pbb of k failures:

$\left(\frac{1}{e}\right)^{c \cdot \log n} \approx \frac{1}{n^{c'}}$

Theorem 12.11. *By using Algorithm 12.10 it is possible to elect a leader w.h.p. in* $\mathcal{O}(\log^2 n)$ *time slots if* n *is not known.*

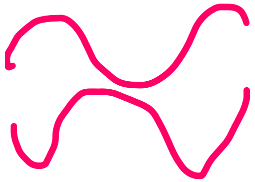
$k \approx \log n$

$c, 2c, 3c, \cancel{3}4c, \ldots, k \cdot c$

$= c \cdot (1+2+3+\ldots+k) = c \cdot \frac{k(k-1)}{2} = O(\log^2 n)$

Algorithm 12.12 Uniform leader election with CD

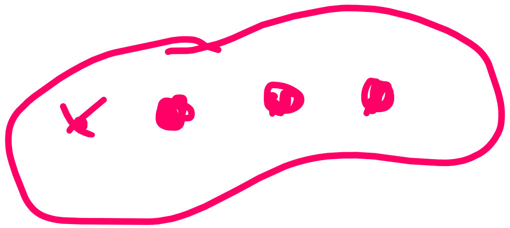
Every node v executes the following code:
repeat
    transmit with probability $\frac{1}{2}$
    if at least one node transmitted **then**
        all nodes that did not transmit quit the protocol
    end if
until one node transmits alone



silence	1 transmits	collision
background noise	message → makes sense	high energy no messages recognized

Algorithm 12.12 Uniform leader election with CD

Every node v executes the following code:
repeat
  transmit with probability $\frac{1}{2}$ $\frac{1}{4}$
  if at least one node transmitted **then**
    all nodes that did not transmit quit the protocol
  end if
until one node transmits alone

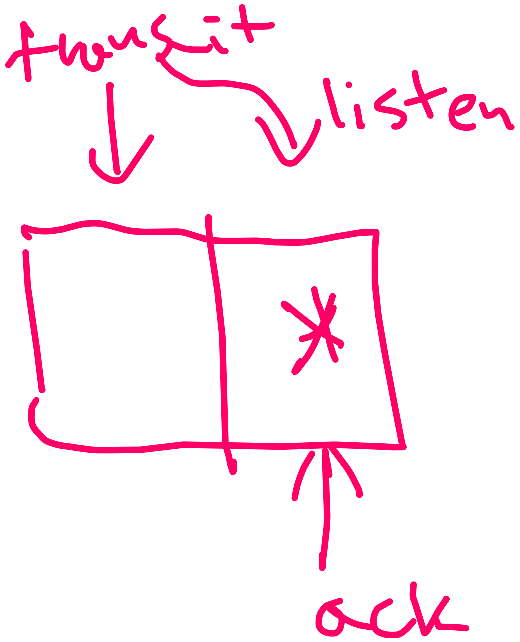


3, 4 $\frac{1}{200}$

trick $n > 1$

transmits

listeners



Theorem 12.13. *With collision detection we can elect a leader using Algorithm 12.12 w.h.p. in* $\mathcal{O}(\log n)$ *time slots.*

k - nodes, 1 round

X - # transmitting nodes

$$\underline{Pr\left[1 \leq X \leq \left\lceil \frac{k}{2} \right\rceil\right]} = P\left[X \leq \left\lceil \frac{k}{2} \right\rceil\right] - Pr[X = 0] \geq \frac{1}{2} - \frac{1}{2^k} \geq \frac{1}{4}.$$

X=0 - no progres

$X > \frac{k}{2}$, no big progress

$\geq$ $\underline{\underline{\log n}}$ steps where $1 \leq X < \left\lceil \frac{k}{2} \right\rceil$, each with ppb $\geq \underline{\underline{\frac{1}{4}}}$

Algorithm 12.14 Fast uniform leader election

$i := 1$
repeat
  $i := 2 \cdot i$
  transmit with probability $1/2^i$
until no node transmitted
{End of Phase 1}
$l := 2^{i/2}$
$u := 2^i$
while $l + 1 < u$ **do**
  $j := \lceil \frac{l+u}{2} \rceil$
  transmit with probability $1/2^j$
  if no node transmitted **then**
    $u := j$
  else
    $l := j$
  end if
end while
{End of Phase 2}
$k := u$
repeat
  transmit with probability $1/2^k$
  if no node transmitted **then**
    $k := k - 1$
  else
    $k := k + 1$
  end if
until exactly one node transmitted

log log n

```
i := 1
repeat
    i := 2 · i
    transmit with probability 1/2^i
until no node transmitted
{End of Phase 1}
```

j

$$2^{2^j} \le n \le 2^{2^{j+1}} = 2^{2^j \cdot 2} = \left(2^{2^j}\right)^2$$

$2^{2^j} \approx n$

pbb $= \frac{1}{n}$

pbb $= \frac{1}{n^2}$

$$p = \frac{1}{n^2}$$

$$1 - \left(1 - \frac{1}{n^2}\right)^n$$

$$= 1 - \left(1 - \frac{1}{n^2}\right)^{n^2/n} \approx$$

$$= 1 - \left(\frac{1}{e}\right)^{1/n} \approx 0$$

[illegible] 22

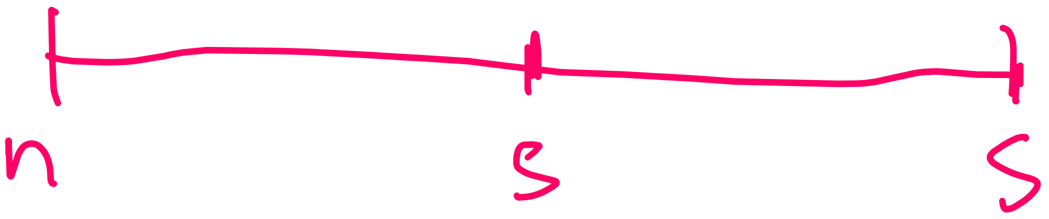
1

```
l := 2^{i/2}
u := 2^i
while l + 1 < u do
    j := ⌈(l+u)/2⌉
    transmit with probability 1/2^j
    if no node transmitted then
        u := j
    else
        l := j
    end if
end while
{End of Phase 2}
```

2^i

$\frac{1}{2^{i/2}}$ $\frac{1}{n}$ $\frac{1}{2^i}$

$\uparrow$ no silence $\uparrow$ silence



$\log(i/2)$

$i(\log\log n)$

$i \approx \log n$

$l \quad o \quad u$

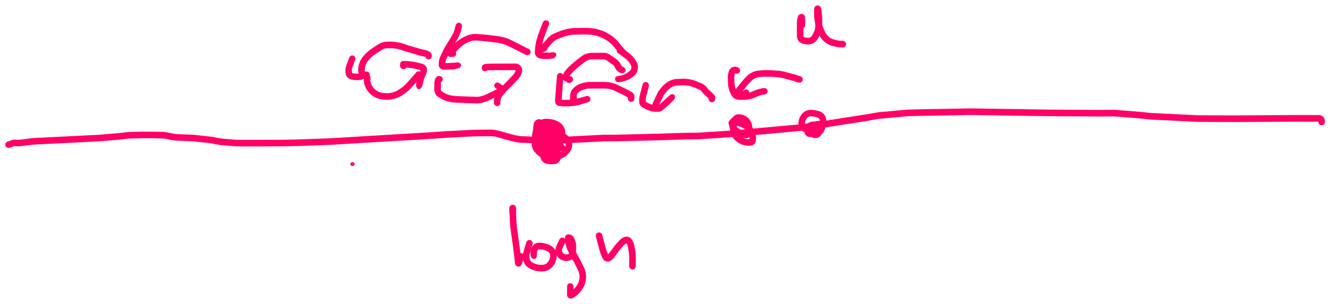
$u = l+1$

```
k := u
repeat
    transmit with probability 1/2^k
    if no node transmitted then
        k := k - 1
    else
        k := k + 1
    end if
until exactly one node transmitted
```

l, u

random walk

pbb. $\frac{1}{2^k} \approx 0$



$\frac{1}{2^k} \gg \frac{1}{n}$

$\log n$

$\binom{n}{1} \cdot \frac{1}{2^n} \cdot \left(1 - \ldots\right)$

Lemma 12.15. *If* $j > \log n + \log\log n$, *then* $Pr[X > 1] \le \frac{1}{\log n}$.

Proof. The nodes transmit with probability $1/2^j < 1/2^{\log n + \log\log n} = \frac{1}{n \log n}$. The expected number of nodes transmitting is $E[X] = \frac{n}{n \log n}$. Using Markov's inequality (see Theorem 12.27) yields $Pr[X > 1] \le Pr[X > E[X] \cdot \log n] \le \frac{1}{\log n}$. $\square$

Lemma 12.16. *If $j < \log n - \log\log n$, then $P[X = 0] \leq \frac{1}{n}$.*

Proof. The nodes transmit with probability $1/2^j > 1/2^{\log n - \log\log n} = \frac{\log n}{n}$. Thus, the probability that a node is silent is at most $1 - \frac{\log n}{n}$. Hence, the probability for a silent time slot, i.e., $Pr[X = 0]$, is at most $(1 - \frac{\log n}{n})^n = e^{-\log n} = \frac{1}{n}$. $\square$

Lemma 12.19. *Let v be such that $2^{v-1} < n \leq 2^v$, i.e., $v \approx \log n$. If $k > v + 2$, then $Pr[X > 1] \leq \frac{1}{4}$.*

Proof. Markov's inequality yields

$$Pr[X > 1] = Pr\left[X > \frac{2^k}{n}E[X]\right] < Pr[X > \frac{2^k}{2^v}E[X]] < Pr[X > 4E[X]] < \frac{1}{4}.$$

□

Lemma 12.20. *If $k < v - 2$, then $P[X = 0] \leq \frac{1}{4}$.*

Proof. A similar analysis is possible to upper bound the probability that a transmission fails if our estimate is too small. We know that $k \leq v - 2$ and thus

$$Pr[X = 0] = \left(1 - \frac{1}{2^k}\right)^n < e^{-\frac{n}{2^k}} < e^{-\frac{2^{v-1}}{2^k}} < e^{-2} < \frac{1}{4}.$$

∎

Lemma 12.21. *If $v - 2 \leq k \leq v + 2$, then the probability that exactly one node transmits is constant.*

Proof. The transmission probability is $p = \frac{1}{2^{v \pm \Theta(1)}} = \Theta(1/n)$, and the lemma follows with a slightly adapted version of Theorem 12.2. $\square$

Lemma 12.22. *With probability* $1-\frac{1}{\log n}$ *we find a leader in phase 3 in* $\mathcal{O}(\log\log n)$ *time.*

Proof. For any k, because of Lemmas 12.19 and 12.20, the random walk of the third phase is biased towards the good area. One can show that in $\mathcal{O}(\log\log n)$ steps one gets $\Omega(\log\log n)$ good transmissions. Let Y denote the number of times exactly one node transmitted. With Lemma 12.21 we obtain $E[Y] = \Omega(\log\log n)$. Now a direct application of a Chernoff bound (see Theorem 12.28) yields that these transmissions elect a leader with probability $1 - \frac{1}{\log n}$. □

Theorem 12.24. *Any uniform protocol that elects a leader with probability of at least* $1 - \frac{1}{2}^t$ *must run for at least* t *time slots.*

$t = \log\log n$

$1 - \left(\frac{1}{2}\right)^{\log\log n} = 1 - \frac{1}{\log n}$

System with two nodes:

$$Pr[X = 1] = 2p \cdot (1 - p) \leq \frac{1}{2}.$$

$\binom{2}{1} \cdot p \cdot (1-p)$

$\left(\frac{1}{2}\right)^t$

Wake-up problem

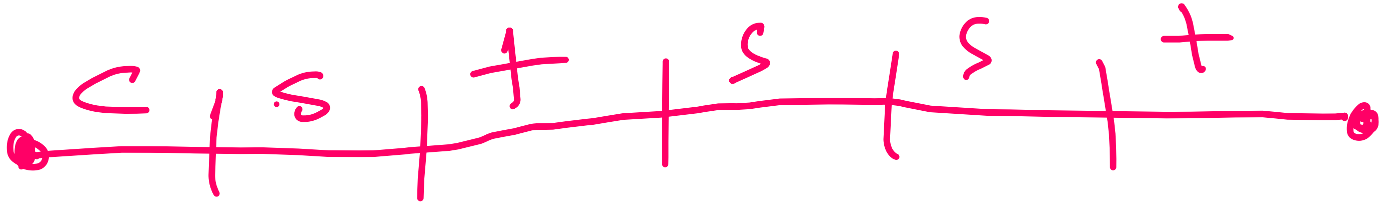
organize waking-up

lower bounds and bad conf.

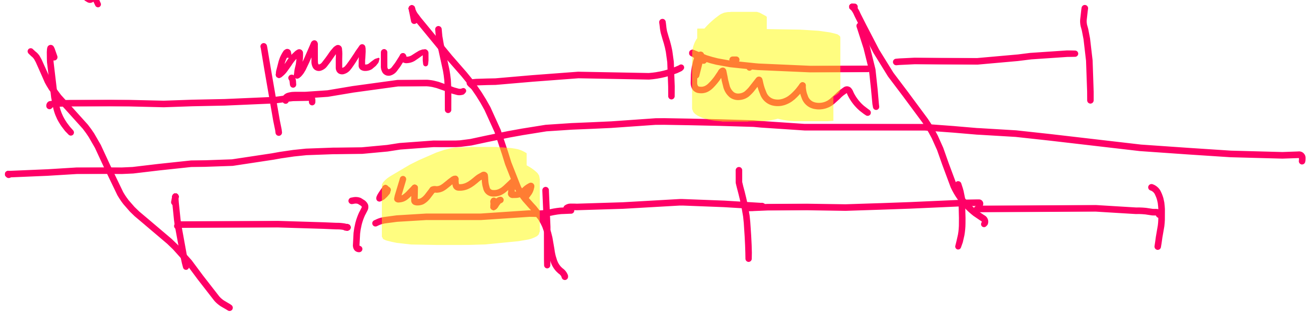
$O(n/\log n)$ to wake up
with failure prob. --- $\frac{1}{\log n}$

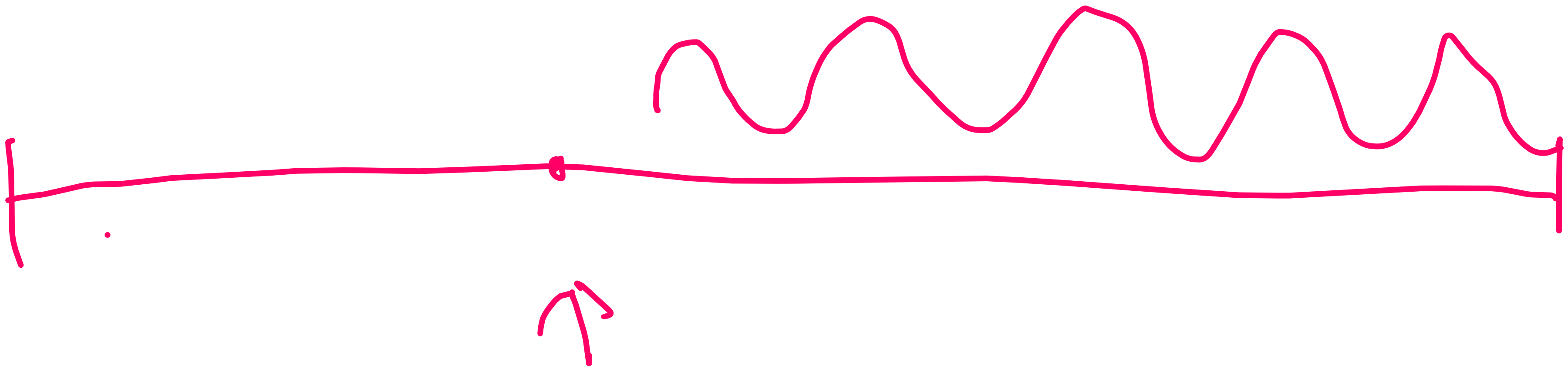
Cai-Wang

slots



no common clock



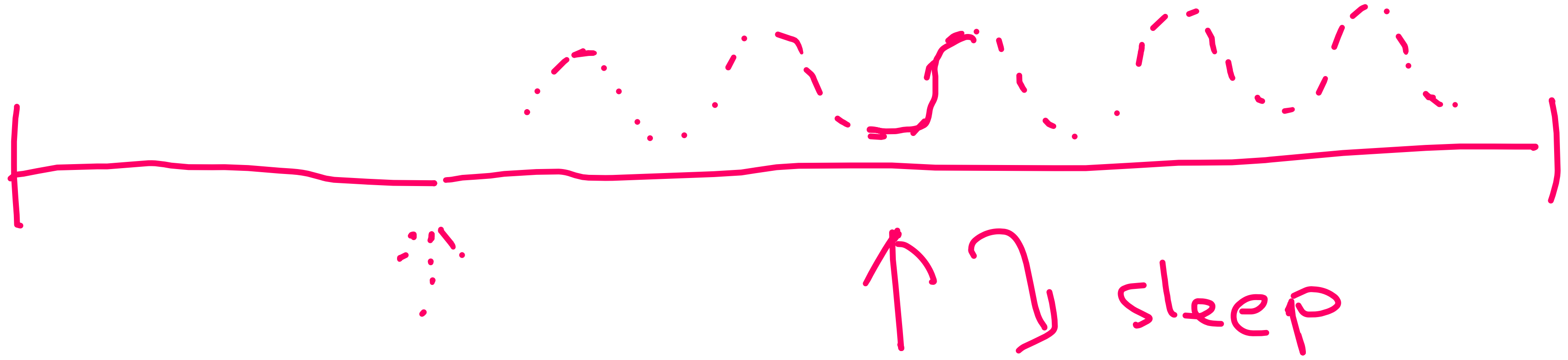


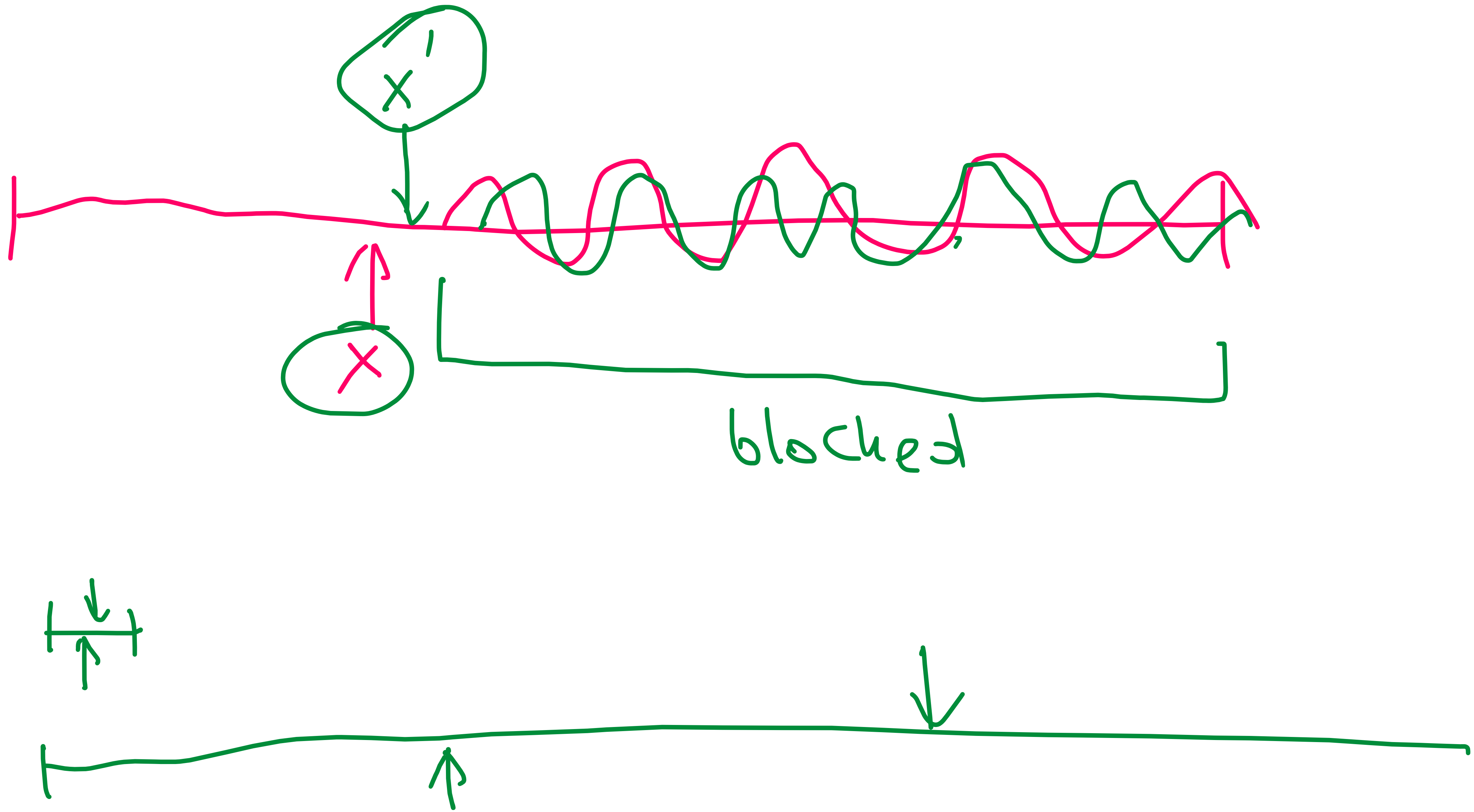
x

listen

if empty then

transmit


sleep


x'
x
blocked